Change Point Detection

Methods Overview

Mikhail Hushchyn (**mhushchyn@hse.ru**)[1]

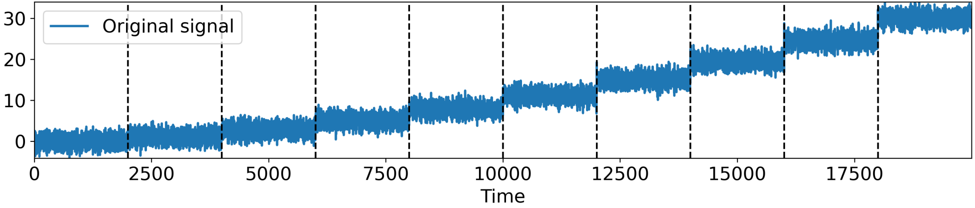
[1] HSE University

LAMBDA • HSE

Introduction

Definition

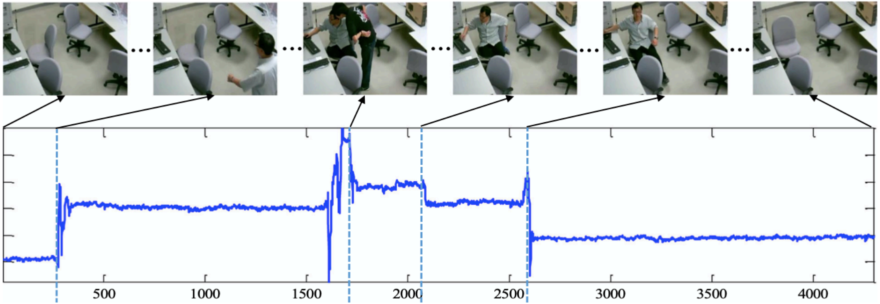
Change Point is a moment of time when a time series changes its behaviour.



Problem: detect all change points in the time series. Labels are unknown.

Video Segmentation

Cases: events detection; automatic summarization;

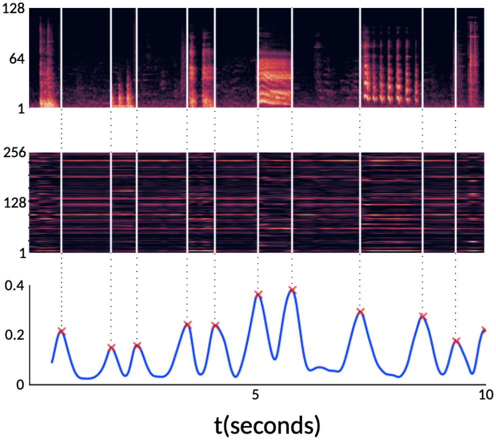


Gao, Z., Lu, G., Lyu, C. et al. Key-frame selection for automatic summarization of surveillance videos: a method of multiple change-point detection. Machine Vision and Applications 29, 1101–1117 (2018).

Sound Segmentation

- Sound events detection:
  - change of musical genre
  - voice changes
  - the emergence of new sounds, etc.
- Segmentation in speech recognition

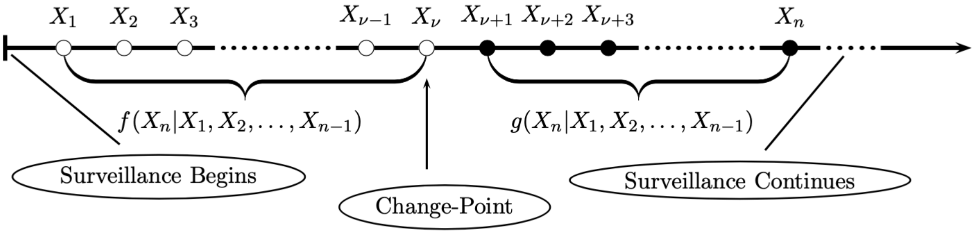
Zhao, Shuyang et al. Active Learning for Sound Event Detection. ArXiv abs/2002.05033 (2020)



Applications

- Quality control
- Faults and anomalies detection
- Computer network surveillance and security
- Finance and economics
- Seismic data processing
- Segmentation of signals and images

Definition



The simplest iid case:

- $f(x_i|x_1, x_2, \ldots, x_{i-1}) = f(x_i) = p_{\theta_0}(x_i)$ is pre-change density for $i \leq v$;
- $g(x_i|x_1, x_2, \ldots, x_{i-1}) = g(x_i) = p_{\theta_1}(x_i)$ is post-change density for $i \geq v + 1$.

Shewhart Control Charts

Hypotheses Test

Take a sample of size N: $x_1, x_2, ..., x_N$. Make a test between the two hypotheses:

$$\mathbf{H}_0 : \theta = \theta_0$$

$$\mathbf{H}_1 : \theta = \theta_1$$

Calculate the **decision function** S_1^N, where:

$$S_j^k = \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(x_i)}{p_{\theta_0}(x_i)}$$

where p_{θ_0} and p_{θ_1} are **known**.

Hypotheses Test

Using the Neyman-Pearson lemma, the optimal **decision rule** d is

$$\mathsf{d} = \begin{cases} 0 & \text{if } \mathsf{S}_1^{\mathsf{N}} < \mathsf{h}\text{; } \mathbf{H}_0 \text{ is chosen} \\ 1 & \text{if } \mathsf{S}_1^{\mathsf{N}} \geq \mathsf{h}\text{; } \mathbf{H}_1 \text{ is chosen} \end{cases}$$

where h is a chosen threshold. **Alarm time** $\mathsf{t_a}$ is defined as

$$\mathsf{t_a} = \mathsf{N} \cdot \min\{\mathsf{K} : \mathsf{d_K} = 1\}$$

where $\mathsf{d_K}$ is the decision rule for the K-th sample.

Example

Case of Gaussian distribution:

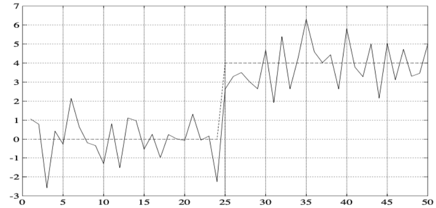
$$p_\theta(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

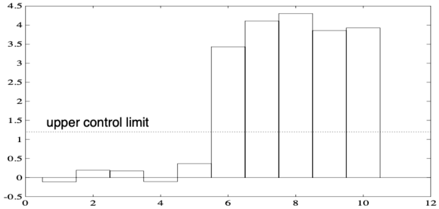
where changing parameter is μ:

$$\mathbf{H}_0 : \mu = \mu_0$$

$$\mathbf{H}_1 : \mu = \mu_1$$

W.A. Shewhart (1931), Economic Control of Quality Manufactured Product, D.Van Nostrand Reinhold, Princeton, NJ.





CUSUM Algorithm

Online case

The **decision function** g_k is

$$g_k = \sup\{0, g_{k-1} + \ln \frac{p_{\theta_1}(x_i)}{p_{\theta_0}(x_i)}\}$$

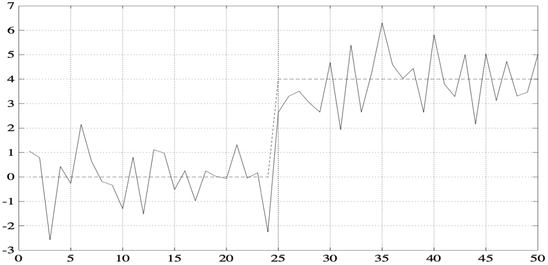
The optimal **decision rule** d_k is

$$d_k = \begin{cases} 0 & \text{if } g_k < h\text{; } \mathbf{H}_0 \text{ is chosen} \\ 1 & \text{if } g_k \geq h\text{; } \mathbf{H}_1 \text{ is chosen} \end{cases}$$

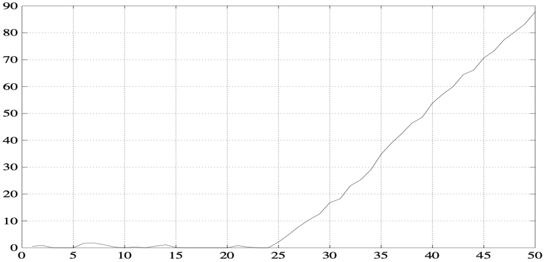
Then, the **alarm time** t_a is

$$t_a = \min\{k : g_k \geq h\}$$

Online case



Gaussian signal with changing mean μ.



Typical behaviour of g_k.

[1] E.S. Page (1954a). Continuous inspection schemes. Biometrika, vol.41, pp.100-115.
[2] **A.N. Shiryaev** (1961). The problem of the most rapid detection of a disturbance in a stationary process. Soviet Math. Dokl., no 2, pp.795-799.

Offline case

Consider a test between the two hypotheses:

$$\begin{aligned} \mathbf{H}_0 &: \theta = \theta_0 \text{ for } 1 \le i \le k \\ 1 \le j \le k,\ \mathbf{H}_j &: \theta = \theta_0 \text{ for } 1 \le i \le j-1 \\ &\quad \theta = \theta_1 \text{ for } j \le i \le k \end{aligned}$$

The **decision function** g_k is

$$g_k = \max_{1 \le j \le k} \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(x_i)}{p_{\theta_0}(x_i)}$$

where p_{θ_0} and p_{θ_1} are **known**.

GLR Algorithm

Generalized Likelihood Ratio (GLR)

Consider a case with **unknown** parameter θ_1 after change. Then, the **decision function** g_k is defined as

$$g_k = \max_{1 \leq j \leq k} \sup_{\theta_1} \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(x_i)}{p_{\theta_0}(x_i)}$$

where θ_0 is **known**.

The **alarm time** t_a is estimated as

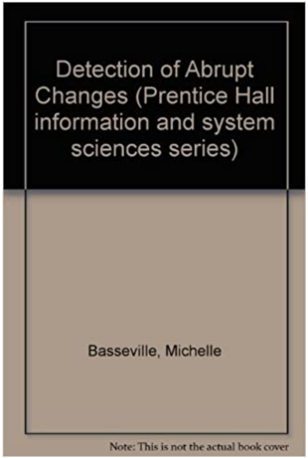
$$t_a = \min\{k : g_k \geq h\}$$

G. Lorden (1971). Procedures for reacting to a change in distribution. Annals Mathematical Statistics, vol.42, pp.1897-1908.

Summary

- Shewhart Control Charts
- CUSUM
- GLR

M. Basseville, I. Nikiforov, Detection of abrupt changes: theory and application, Prentice Hall Englewood Cliffs, 1993.



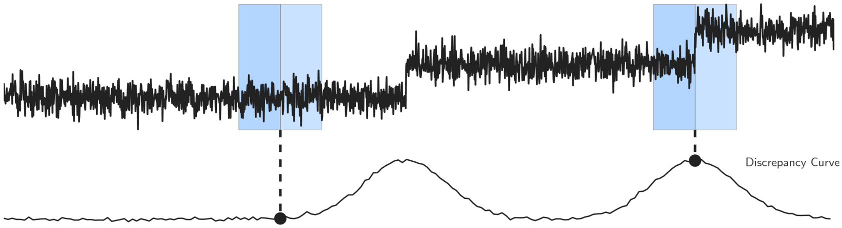
Direct Density Ratio Estimation

Motivation

- It is hard to estimate $p_{\theta_0}(x)$ and $p_{\theta_1}(x)$
- It is even harder in multidimensional case
- Can we estimate $\frac{p_{\theta_1}(x)}{p_{\theta_0}(x)}$ directly from data?

Rolling Windows

- Left window is **reference** with density $p_{rf}(x_i)$ for $k - n_{rf} < i \leq k$;
- Right window is **test** with density $p_{te}(x_i)$ for $k < i \leq k + n_{te}$.



KLIEP Algorithm

Test between the two hypotheses:

$$\begin{aligned}\mathbf{H}_0 &: p(x_i) = p_{rf}(x_i) \text{ for } k - n_{rf} < i \leq k + n_{te} \\ \mathbf{H}_1 &: p(x_i) = p_{rf}(x_i) \text{ for } k - n_{rf} < i \leq k \\ &\quad p(x_i) = p_{te}(x_i) \text{ for } k < i \leq k + n_{te}\end{aligned}$$

The **decision function** g_k is

$$g_k = \sum_{i=k+1}^{k+n_{te}} \ln \frac{p_{te}(x_i)}{p_{rf}(x_i)} = \sum_{i=k+1}^{k+n_{te}} \ln w(x_i)$$

where $w(x_i) = \frac{p_{te}(x_i)}{p_{rf}(x_i)}$

KLIEP Algorithm

Kullback-Leibler Importance Estimation Procedure (KLIEP) [1] supposes:

$$\hat{w}(x_j) = \sum_{i=k+1}^{k+n_{te}} \alpha_i K(x_j, x_i)$$

$$K(x_j, x_i) = \exp\left(-\frac{||x_j - x_i||_2^2}{2\sigma^2}\right)$$

where $\{\alpha_i\}$ are the parameters to be fitted from samples.

[1] M. Sugiyama et al. (2008), Direct importance estimation for covariate shift adaptation. Annals of the Institute of Statistical Mathematics, 60(4): 699–746

KLIEP Algorithm

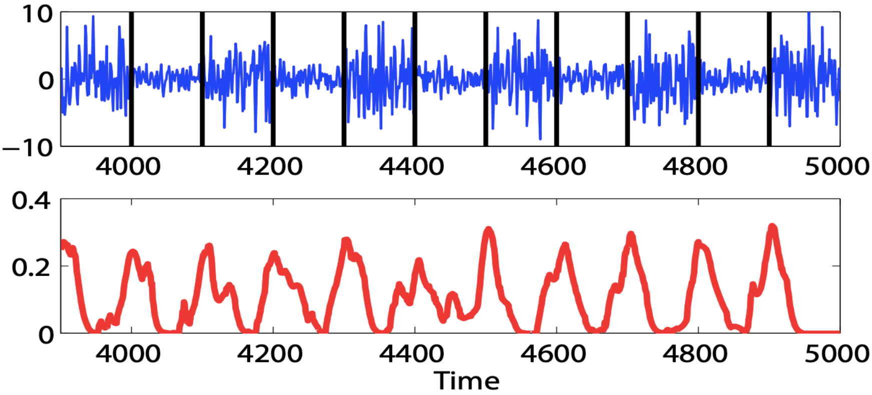
The optimization procedure:

$$\sum_{j=k+1}^{k+n_{te}} \ln\Big(\sum_{i=k+1}^{k+n_{te}} \alpha_i K(x_j, x_i)\Big) \to \max_{\alpha_i}$$

$$\frac{1}{n_{rf}} \sum_{j=k-n_{rf}+1}^{k} \sum_{i=k+1}^{k+n_{te}} \alpha_i K(x_j, x_i) = 1$$

$$\alpha_i \geq 0$$

Example



S. Liu et al. (2013), Change-point detection in time-series data by relative density-ratio estimation, Neural Networks, V 43, pp 72-83

Similar Approaches

- Logistic Regression Method
- Kernel Mean Matching (KMM)
- Kullback-Leibler Importance Estimation Procedure (KLIEP)
- Unconstrained Least-Squares Importance Fitting (uLSIF)
- Relative uLSIF (RuLSIF)

S. Liu et al. (2013), Change-point detection in time-series data by relative density-ratio estimation, Neural Networks, V 43, pp 72-83

Discrepancy Based Methods

Sliding Windows

Idea: let's compare observations in two sliding windows using different cost functions.

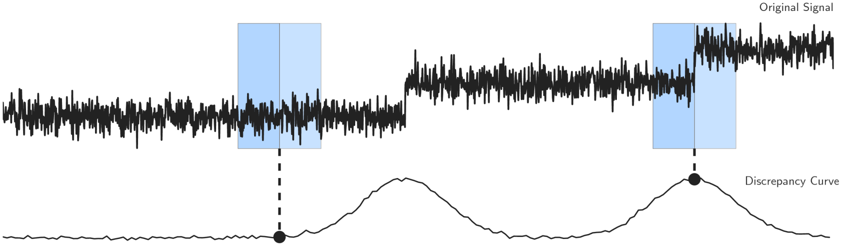


Рис.: **https://centre-borelli.github.io/ruptures-docs/detection/window.html**

Discrepancy

Consider a time series of observations $x_k \in \mathcal{R}^d$:

$$x_1, x_2, x_3, ..., x_i, x_{i+1}, x_{i+2}, \ldots$$

Then, for two windows $x_{u..v}$ and $x_{v..w}$ **discrepancy** between them is defined as [1]:

$$d(x_{u..v}, x_{v..w}) = c(x_{u..w}) - c(x_{u..v}) - c(x_{v..w}),$$

where $c(\cdot)$ is a **cost function**, $u < v < w$.

[1] C. Truong, L. Oudre, N. Vayatis (2020). Selective review of offline change point detection methods. Signal Processing, 167:107299.

Cost Functions

Detecting a shift in the median of a signal:

$$c(x_I) = \sum_{t \in I} ||x_t - \bar{x}||_1$$

Detecting a shift in the mean of a signal:

$$c(x_I) = \sum_{t \in I} ||x_t - \bar{x}||_2$$

Detecting the shift of the mean and variance of a Gaussian signal:

$$c(x_{a..b}) = (b - a) \log \det \hat{\Sigma} + \sum_{t=a+1}^{b} (x_t - \bar{x})^T \hat{\Sigma}^{-1} (x_t - \bar{x})$$

Cost Functions

Generalized Likelihood Ratio (GLR) test:

$$c(x_{a..b}) = -\sup_{\theta} \sum_{t=a+1}^{b} \ln p_{\theta}(x_i)$$

Detection of many types of signal change points:

$$c(x_{a..b}) = \sum_{t=a+1}^{b} k(x_t, x_t) - \frac{1}{b-a} \sum_{s,t=a+1}^{b} k(x_s, x_t),$$

where $k(x_s, x_t)$ is a kernel.

Windows Algorithm

Algorithm 3 Algorithm `Win`

Input: signal $\{y_t\}_{t=1}^T$, cost function $c(\cdot)$, half-window width w, peak search procedure `PKSearch`.

Initialize $Z \leftarrow [0, 0, \dots]$ a T-long array filled with 0. ▷ Score list.

for $t = w, \dots, T - w$ **do**

    $p \leftarrow (t - w)..t$.

    $q \leftarrow t..(t + w)$.

    $r \leftarrow (t - w)..(t + w)$.

    $Z[t] \leftarrow c(y_r) - [c(y_p) + c(y_q)]$.

end for

$L \leftarrow$ `PKSearch`(Z) ▷ Peak search procedure.

Output: set L of estimated breakpoint indexes.

Example

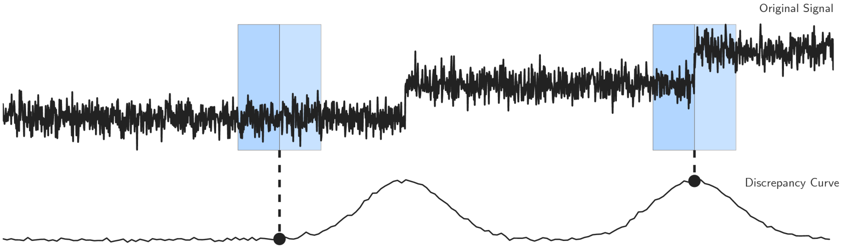


Рис.: **https://centre-borelli.github.io/ruptures-docs/detection/window.html**

Binary Segmentation Algorithm

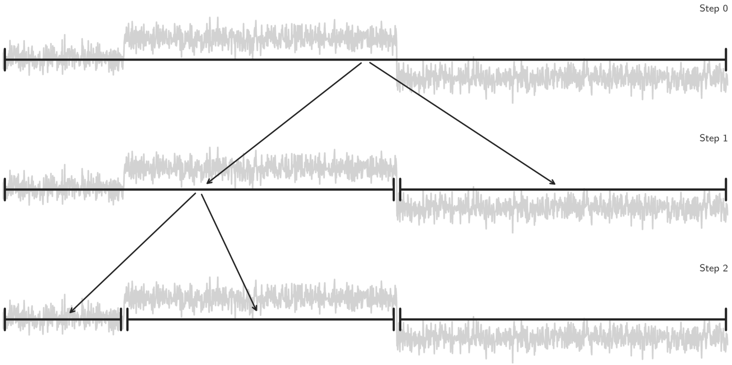


Рис.: **https://centre-borelli.github.io/ruptures-docs/user-guide/detection/binseg/**

Binary Segmentation Algorithm

Algorithm 4 Algorithm `BinSeg`

Input: signal $\{y_t\}_{t=1}^{T}$, cost function $c(\cdot)$, stopping criterion.
Initialize $L \leftarrow \{\ \}$. ▷ Estimated breakpoints.
repeat
  $k \leftarrow |L|$. ▷ Number of breakpoints
  $t_0 \leftarrow 0$ and $t_{k+1} \leftarrow T$ ▷ Dummy variables.
  if $k > 0$ **then**
    Denote by t_i $(i = 1, \ldots, k)$ the elements (in ascending order) of L, ie $L = \{t_1, \ldots, t_k\}$.
  end if
  Initialize G a $(k+1)$-long array. ▷ list of gains
  for $i = 0, \ldots, k$ **do**

$$G[i] \leftarrow c(y_{t_i..t_{i+1}}) - \min_{t_i < t < t_{i+1}} [c(y_{t_i..t}) + c(y_{t..t_{i+1}})] \ .$$

  end for
  $\widehat{i} \leftarrow \operatorname{argmax}_i G[i]$
  $\hat{t} \leftarrow \operatorname{argmin}_{t_{\widehat{i}} < t < t_{\widehat{i}+1}} [c(y_{t_{\widehat{i}}..t}) + c(y_{t..t_{\widehat{i}+1}})]$.
  $L \leftarrow L \cup \{\hat{t}\}$
until stopping criterion is met.
Output: set L of estimated breakpoint indexes.

Similar Approaches

- Binary segmentation (Binseg)
- Window-based change point detection (Windows)
- Bottom-up segmentation (BottomUp)
- Dynamic programming
- Linearly penalized segmentation (Pelt)
- Kernel change point detection

[1] C. Truong, L. Oudre, N. Vayatis. Selective review of offline change point detection methods. Signal Processing, 167:107299, 2020.
[2] Implementation: **https://centre-borelli.github.io/ruptures-docs/**

Literature

1 M. Basseville, I. Nikiforov, Detection of abrupt changes: theory and application, 104, Prentice Hall Englewood Cliffs, 1993.

2 A. Tartakovsky, I. Nikiforov, and M. Basseville. Sequential Analysis: Hypothesis Testing and Changepoint Detection. Monographs on Statistics and Applied Probability. Chapman and Hall/CRC Press, Boca Raton, Florida, 2014.

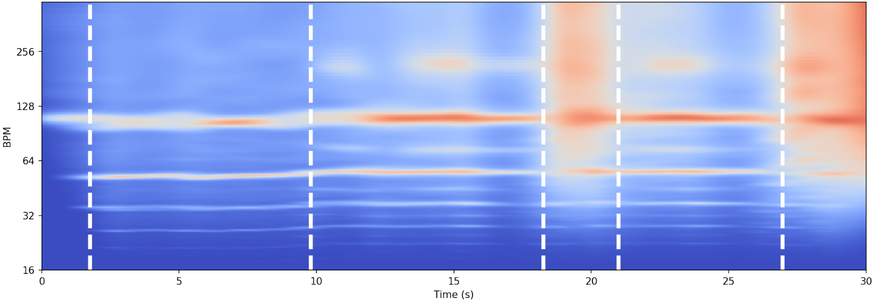
3 S. Liu, M. Yamada, N. Collier, M. Sugiyama, Change-point detection in time-series data by relative density-ratio estimation, Neural Networks 43 (2013) 72 – 83.

4 S. Aminikhanghahi, D. J. Cook, A survey of methods for time series change point detection, Knowledge and Information Systems 51 (2) (2017) 339 – 367.

5 C. Truong, L. Oudre, N. Vayatis. Selective review of offline change point detection methods. Signal Processing, 167:107299, 2020.

Conclusion

P. I. Tchaikovsky, The Dance of the Sugar Plum Fairy:



Interactive demo:

https://centre-borelli.github.io/ruptures-docs/examples/music-segmentation/